$$\mathbf{X} = \begin{bmatrix} 4 & 10 \\ 8 & 15 \\ 6 & 12 \end{bmatrix}$$

- Column-wise Mean:
  - For the first column: $\mu_1 = \frac{4+8+6}{3} = 6$
  - For the second column: $\mu_2 = \frac{10+15+12}{3} = 12.33$

  So, the column-wise mean vector is $\mu = [6, 12.33]$.
- Row-wise Mean:
  - For the first row: $\mu_1 = \frac{4+10}{2} = 7$
  - For the second row: $\mu_2 = \frac{8+15}{2} = 11.5$
  - For the third row: $\mu_3 = \frac{6+12}{2} = 9$

  The row-wise mean vector is $\mu = [7, 11.5, 9]$.

Column Standardization

Column standardization is a preprocessing technique used in machine learning to scale each feature (column) so that it has a **mean of 0** and a **standard deviation of 1**. This transformation, also called **Z-score standardization** or **Z-score normalization**, makes data values comparable across different features. Standardization is especially helpful in algorithms sensitive to the scale of features, such as linear regression, logistic regression, and neural networks.

How Column Standardization Works

Given a data matrix Xwith m observations (rows) and n features (columns), we standardize each feature j as follows:

For each feature (column) j:

1. **Calculate the Mean (μj)**: Find the average value of all entries in column j.
2. **Calculate the Standard Deviation (σj)**: Measure the spread of values around the mean for column j.
3. **Apply Standardization Formula**:

$$z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j}$$

After standardization:

- Each column will have a mean of 0.
- Each column will have a standard deviation of 1.

Example of Column Standardization

Consider a simple data matrix with two features, **Age** and **Income**:

$$\mathbf{X} = \begin{bmatrix} 20 & 30000 \\ 30 & 50000 \\ 40 & 70000 \end{bmatrix}$$

Step 1: Calculate Mean and Standard Deviation

- For Age:
  - Mean $\mu_{\text{Age}} = \frac{20+30+40}{3} = 30$
  - Standard Deviation $\sigma_{\text{Age}} = \sqrt{\frac{(20-30)^2+(30-30)^2+(40-30)^2}{3}} = 10$
- For Income:
  - Mean $\mu_{\text{Income}} = \frac{30000+50000+70000}{3} = 50000$
  - Standard Deviation $\sigma_{\text{Income}} = \sqrt{\frac{(30000-50000)^2+(50000-50000)^2+(70000-50000)^2}{3}} = 20000$

Step 2: Apply Standardization Formula

For each value, subtract the mean and divide by the standard deviation:

$$\mathbf{X}_{\text{standardized}} = \begin{bmatrix} \frac{20-30}{10} & \frac{30000-50000}{20000} \\ \frac{30-30}{10} & \frac{50000-50000}{20000} \\ \frac{40-30}{10} & \frac{70000-50000}{20000} \end{bmatrix} = \begin{bmatrix} -1 & -1 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}$$

When to Use Column Standardization

- **Algorithms Sensitive to Scale**: Many machine learning algorithms (e.g., gradient-based optimizations, k-means clustering) perform better with standardized data.
- **When Data is Not Normally Distributed**: Standardization brings features to a common scale without distorting differences in variances, even if the original data is not normally distributed.
- **Feature Comparability**: When features have different units or scales, standardization ensures that each feature contributes equally.

Covariance of a Data Matrix

In a data matrix, **covariance** measures the degree to which two features (variables) vary together. For any two features, a positive covariance indicates that as one feature increases, the other tends to increase, while a negative covariance indicates that as one feature increases, the other tends to decrease. When the covariance is close to zero, it suggests that the features are independent or uncorrelated.

Covariance is useful in understanding relationships between features and is especially valuable in techniques like **Principal Component Analysis (PCA)** for dimensionality reduction.

Covariance Matrix

For a dataset with m observations (rows) and n features (columns), we can calculate an **n×n covariance matrix**, denoted as Σ, where each element σij represents the covariance between features i and j.

Covariance Formula

For two features X and Y, the covariance is calculated as:

$$\text{cov}(X, Y) = \frac{1}{m-1} \sum_{k=1}^{m} (x_k - \mu_X)(y_k - \mu_Y)$$

Where:

- xk are values of features X and Y for the k-th observation.
- μX are the means of features X and Y.
- m is the total number of observations.